

BUNDESREPUBLIK DEUTSCHLAND



| | |
|-------------------|-----|
| REC'D 23 JAN 2004 | |
| WIPO | PCT |

**Prioritätsbescheinigung über die Einreichung
einer Patentanmeldung**

Aktenzeichen: 102 52 230.8

Anmeldetag: 11. November 2002

Anmelder/Inhaber: Robert Bosch GmbH, Stuttgart/DE

Bezeichnung: Verfahren zur Übertragung von Daten

IPC: H 04 L, B 60 R

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 24. November 2003
Deutsches Patent- und Markenamt
Der Präsident
im Auftrag

Siegel

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

5 07.11.2002

1. Robert Bosch GmbH, 70442 Stuttgart
2. ZF Lenksysteme GmbH, Richard-Bullinger-Str. 77
73527 Schwäbisch Gmünd

10 Verfahren zur Übertragung von Daten

Die vorliegende Erfindung betrifft ein Verfahren zur
Übertragung von Daten, bei dem nach einem vorgebbaren
15 Signaturbildungsverfahren eine erste Signatur in
Abhängigkeit der zu übertragenden Daten gebildet wird. Die
erste Signatur wird zusammen mit den Daten in sogenannten
Botschaften übertragen. Nach dem Signaturbildungsverfahren
wird eine zweite Signatur in Abhängigkeit von den
20 übertragenen Daten gebildet und die erste Signatur mit der
zweiten Signatur zum Zwecke einer Signaturanalyse
verglichen.

Die Erfindung betrifft außerdem ein Computerprogramm, das
auf einem Steuergerät, insbesondere auf einer Recheneinheit
ablauffähig ist.

Schließlich betrifft die vorliegende Erfindung auch ein
Steuergerät für ein Kraftfahrzeug. Das Steuergerät umfasst
30 eine Recheneinheit, die insbesondere als ein Mikroprozessor
ausgebildet ist, und ein Speicherelement, das insbesondere
als ein Schreib-Lese-Speicher mit wahlfreiem Zugriff (RAM),
als ein Nur-Lese-Speicher (ROM) oder als ein Flash-Speicher
(Flash-Memory) ausgebildet ist. Auf dem Speicherelement ist
35 ein Computerprogramm abgelegt, das auf der Recheneinheit

ablauffähig ist.

Stand der Technik

5

Aus dem Stand der Technik ist die Signaturanalyse als eine Methode zur Datenkomprimierung bekannt, die auf der sogenannten CRC (Cyclic Redundancy Check)-Methode basiert. In Abhängigkeit der zu übertragenden Daten wird nach einem vorgebbaren Signaturbildungsverfahren ein resultierendes Datenwort gebildet, welches die Signatur darstellt und den komprimierten Datenstrom charakterisiert. Die Wahrscheinlichkeit, dass trotz Fehlern in dem Datenstrom eine richtige Signatur gebildet wird, ist sehr gering und nimmt mit der Datenbreite des Signaturwortes ab. Signaturen werden beispielsweise eingesetzt für die Sicherung von Datenübertragungen, zur Überwachung von Festwertspeichern von Computern und zur Identifizierung von Datenströmen, wie zum Beispiel von digitalen Hardware-Signalen im Rahmen eines Tests von Schaltungen oder von Daten, die ein Schriftstück enthält. Zum Stand der Technik wird verwiesen auf Abramovici, Miron, et al.: Digital Systems Testing and Testable Design, New York, IEEE Press, 1990.

5

In Hardware wird die Signatur üblicherweise sequentiell Bit für Bit mit Hilfe eines linear rückgekoppelten Schieberegisters (LFSR; Linear Feedback Shift Register) gebildet. Es ist bekannt, dieses sequentielle Signaturanalyse-Verfahren auch in Software zu realisieren.

30

Dieses Verfahren ist aber wegen der sequentiellen Arbeitsweise entweder sehr langsam oder braucht relativ viel Speicherplatz für Tabellen (sogenannte Lookup-Tables), um die Berechnungszeit zu verkürzen. Zum Stand der Technik von solchen softwaremäßig realisierten sequentiellen Verfahren zur Signaturanalyse wird auf Williams, R. N.: A

Painless Guide to CRC Error Detection Algorithms, Version 3, 19.08.1993, Rocksoft Pty Ltd., Hazelwood Park 5066, Australia, ftp://ftp.rocksoft.com/papers/crc_v3.txt verwiesen.

5

Des weiteren ist ein Verfahren zur Signaturbildung mit einer parallelen Arbeitsweise (Datenwort für Datenwort) bekannt. Dieses bekannte Verfahren arbeitet beispielsweise mit Hilfe eines Signaturregisters mit mehreren Eingängen (MISR; Multiple Input Shift Register). Durch die parallele Arbeitsweise können zum Beispiel mit einem 32 Bit-Prozessor 32 Bit Eingangsdaten in einem Schritt verarbeitet werden. Ein derartiges Verfahren ist beispielsweise aus der US 5,938,784 bekannt.

15

Gerade für Sicherheitsanwendungen in einem Kraftfahrzeug, wie beispielsweise Steer-by-Wire oder autonomes Fahren, werden fehlertolerante und damit redundante Elektroniksysteme eingesetzt, die einen hohen Bedarf an sicherem Datenaustausch zwischen Systemen, Steuergeräten, Rechnern und/oder Recheneinheiten haben. Diese Kommunikation kann in drei Kommunikationsebenen erfolgen. Eine System-Kommunikation zwischen zwei oder mehr Steuergeräten erfolgt bspw. über ein Bussystemen, wie beispielsweise CAN (Controller Area Network), TTCAN (Time Triggered CAN), FlexRay oder TTP/C (Time Triggered Protocol Class C). Eine Steuergeräte-Steuergeräte-Kommunikation bzw. eine Rechner-Rechner-Kommunikation, an der zwei Steuergeräte bzw. Rechner beteiligt sind, erfolgt über Schnittstellen, wie beispielsweise RS232, RS422, SPI (Serial Peripheral Interface) oder SCI (Serial Communications Interface). Eine Recheneinheit-Recheneinheit-Kommunikation erfolgt innerhalb eines Steuergerätes. Da bei diesen Systemen neben der Rechenzeit die Datenkommunikationszeit einen Engpass darstellt, ist

eine Verwendung des MISR-Verfahren zur Absicherung der vielen Datenkommunikationen von besonderem Vorteil. Es reduziert die erforderliche Rechenzeit zur Absicherung der Datenkommunikation bei Sender und Empfänger auf ein
5 Minimum.

Bei den bekannten Verfahren kann es - wenn auch äußerst selten - vorkommen, dass trotz fehlerhafter Daten innerhalb einer Botschaft eine fehlerfreie Signatur gebildet wird.

10 Das heißt, dass die zu übertragenden Daten und die übertragenen Daten zwar nicht miteinander übereinstimmen, aber dennoch die erste Signatur, welche über die zu übertragenden Daten gebildet wurde, und die zweite Signatur, welche über die übertragenen Daten gebildet
15 wurde, übereinstimmen. Dies wird als Fehlermaskierung bezeichnet.

Der vorliegenden Erfindung liegt die Aufgabe zu Grunde, bei der Überwachung der Übertragung von Daten ein Verfahren
20 bereit zu stellen, das eine schnelle Bildung einer Signatur erlaubt und die Fehlermaskierungswahrscheinlichkeit reduziert.

5 Zur Lösung dieser Aufgabe schlägt die vorliegende Erfindung ausgehend von dem Verfahren der eingangs genannten Art vor, dass die zu übertragenden Daten invertiert werden, nach dem vorgebbaren Signaturbildungsverfahren in Abhängigkeit der Daten und der invertierten Daten die erste Signatur gebildet wird, die erste Signatur und die Daten übertragen
30 werden, die übertragenen Daten invertiert werden, in Abhängigkeit dieser invertierten Daten und der übertragenen Daten nach dem Signaturbildungsverfahren die zweite Signatur gebildet wird und die erste Signatur mit der zweiten Signatur verglichen wird.

Vorteile der Erfindung

Erfindungsgemäß wird die Signatur nicht nur über die zu übertragenden bzw. übertragenen Daten, sondern auch über die invertierten Daten gebildet. Durch die zusätzliche Verarbeitung der invertierten Daten kann die Fehlermaskierungswahrscheinlichkeit deutlich reduziert werden. Der zusätzliche Aufwand für die Verarbeitung der invertierten Daten ist gering, so dass nach wie vor kurze Rechenzeiten erzielt werden können. Das erfindungsgemäße Verfahren eignet sich deshalb insbesondere zur Anwendung in Sicherheitsanwendungen in einem Kraftfahrzeug.

Das erfindungsgemäße Verfahren kann sowohl hardwaremäßig als auch softwaremäßig realisiert sein. Die softwaremäßige Realisierung hat den Vorteil geringerer Kosten und eines geringeren Gewichtes, da keine Hardware-Bauteile für die Schaltung erforderlich sind. Außerdem bringt eine softwaremäßige Realisierung eine größere Flexibilität (kann ohne großen Aufwand umprogrammiert und an neue Anforderungen angepasst werden).

Gemäß einer vorteilhaften Weiterbildung der Erfindung wird vorgeschlagen, dass mittels eines Signaturregisters mit mehreren Eingängen (MISR; Multiple Input Shift Register) die erste Signatur und/oder die zweite Signatur bitparallel (wortweise) gebildet werden. Dadurch wird die Geschwindigkeit der Signaturbildung erhöht.

In einer bevorzugten Ausführungsform der Erfindung werden die erste Signatur und/oder die zweite Signatur über die Daten mehrerer Botschaften gebildet. Es werden also nicht die Daten jeder übertragene Botschaft mit einer eigenen Signatur individuell abgesichert, sondern es werden die

Daten mehrerer Botschaften gemeinsam abgesichert. Damit wird erreicht, dass das Datenvolumen der für die Signaturanalyse zu übertragenden Daten reduziert wird.

- 5 Dabei wird vorteilhafterweise eine Signatur auf mehrere Botschaften verteilt übertragen. Diese sogenannte Methode der verteilten Signaturübertragung bei der die gemeinsame Signatur auf die Botschaften verteilt wird, die die abzusichernden Daten enthalten, stellt eine zusätzliche
- 10 Möglichkeit zur Reduzierung des zu übertragenden Datenvolumens dar. Ein Steuergerät überträgt beispielsweise auf einem CAN-Bus pro Systemzyklus drei Botschaften mit je maximal acht Byte, die jeweils mit einer 32 Bit-Signatur abgesichert werden sollen. Wenn die Daten bei der
- 15 Empfangseinheit erst nach dem Empfang aller drei Botschaften weiterverarbeitet werden, macht es Sinn, die Daten der Botschaften nicht einzeln mit 32 Bit, sondern gemeinsam mit 32 Bit abzusichern und die 32 Bit (gleich vier Byte) Signaturdaten derart zu verteilen, dass die
- 20 Botschaften optimal mit Daten ausgelastet sind. Hierdurch wird einerseits Datenvolumen für die Datenübertragung reduziert (vier Byte Signatur anstatt 12 Byte Signatur) und andererseits ist die übertragene Signatur nicht so anfällig gegen eine Datenverfälschung (Maskierung) durch eine
- 25 Störung, da die Signatur nicht als ein Block, sondern als drei Teilblöcke zusammen mit den einzelnen Datenworten zu unterschiedlichen Zeitpunkten übertragen wird.

- 30 Bei sicherheitsrelevanten Systemen wie Steer-by-Wire, autonomes Fahren, automatische Spurführung sowie Systemen zur Fahrdynamikregelung über ein Verbundsystem aus Bremse, Lenkung, Fahrwerk und so weiter, werden aus Sicherheitsgründen Mehrrechnersysteme eingesetzt. Dabei werden zwei, drei oder vier Recheneinheiten pro Steuergerät
- 35 eingesetzt. Bei Flugzeugen und in der Raumfahrt können

sogar Systeme mit fünf und mehr Recheneinheiten pro Steuergerät eingesetzt werden. Ein Grundprinzip dieser Mehrrechnersysteme ist, dass vor Beginn von Berechnungen alle Eingangsdaten unter den Recheneinheiten ausgetauscht

5 und danach die Startdaten für die Berechnung bitgenau abgestimmt werden (sogenanntes Eingangsvoting). Die Berechnungsergebnisse werden ebenfalls zwischen allen Recheneinheiten ausgetauscht und auf Bitgenauigkeit verglichen (sogenanntes Ausgangsvoting).

10

Sind die zu übertragenden Daten bitgenaue Eingangsdaten, die beispielsweise in Botschaften über Datenbusse zu den Recheneinheiten gelangen oder Berechnungsergebnisse, die parallel auf mehreren Rechnern redundant erzeugt werden, so 15 werden vorzugsweise zur Überprüfung einer Übereinstimmung dieser Daten lediglich die entsprechenden Signaturen übertragen.

20 Insbesondere unter Einsatz des besonders schnellen und zuverlässigen erfindungsgemäßen Verfahrens zur Signaturbildung kann damit das zu übertragende Datenvolumen reduziert und eine Verarbeitungsgeschwindigkeit erhöht werden.

25 Vorteilhafterweise wird das erfindungsgemäße Verfahren zur Überprüfung des Speicherinhaltes eines Nur-Lese-Speichers, Flash-Speichers oder eines Schreib-Lese-Speichers verwendet. Dies ermöglicht ein effizientes Überprüfen der Datenkonsistenz des Speicherinhaltes.

30

35 Vorzugsweise werden bei dieser erfindungsgemäßen Verwendung die Daten des zu überprüfenden Speicherinhaltes invertiert, nach dem vorgebbaren Signaturbildungsverfahren in Abhängigkeit der zu überprüfenden Daten und der invertierten Daten die erste Signatur gebildet und in einem

Speicherbereich eines Nur-Lese-Speichers, Flash-Speichers oder eines Schreib-Lese-Speichers als eine Sollsingatur abgespeichert. Für eine Überprüfung der sich in dem zu überprüfenden Speicherbereich befindlichen Daten, werden 5 diese Daten invertiert, in Abhängigkeit dieser invertierten Daten und der Daten nach dem Signaturbildungsverfahren die zweite Signatur gebildet und mit der abgespeicherten Sollsingatur verglichen. Es wird also eine erste Signatur, die sogenannte Sollsingatur, über einen zu überprüfenden 10 Speicherinhalt einmalig erzeugt und abgespeichert. Soll dieser Speicherinhalt dann beispielsweise zu einem späteren Zeitpunkt überprüft werden, so wird über den aktuellen Speicherinhalt eine zweite Signatur erzeugt und mit der abgespeicherten Sollsingatur verglichen. Stimmen diese 15 Signaturen nicht überein, kann davon ausgegangen werden, dass sich der Inhalt des zu überwachenden Speicherbereiches verändert hat. Dies erlaubt eine besonders effiziente Überprüfung eines Speicherbereiches auf Datenkonsistenz, da zu einer Überprüfung der Daten lediglich die Signaturen 20 verglichen werden müssen. Außerdem muss die Sollsingatur nur einmal gebildet werden.

Von besonderer Bedeutung ist die Realisierung des erfindungsgemäßen Verfahrens in der Form eines Computerprogramms. Dabei ist das Computerprogramm auf einem Steuergerät, insbesondere auf einer Recheneinheit ablauffähig und zur Ausführung des erfindungsgemäßen Verfahrens geeignet. In diesem Fall wird also die Erfindung durch das Computerprogramm realisiert, so dass dieses 30 Computerprogramm in gleicher Weise die Erfindung darstellt wie das Verfahren, zu dessen Ausführung das Computerprogramm geeignet ist. Das Computerprogramm ist vorzugsweise auf einem Speicherelement abgespeichert. Als Speicherelement kann insbesondere ein elektrisches 35 Speichermedium zur Anwendung kommen, beispielsweise ein

Schreib-Lese-Speicher mit wahlfreiem Zugriff (RAM; Random-Access-Memory), ein Nur-Lese-Speicher (ROM; Read-Only-Memory) oder ein Flash-Speicher (Flash-Memory).

5 Als eine weitere Lösung der Aufgabe der vorliegenden Erfindung wird ausgehend von dem Steuergerät der eingangs genannten Art vorgeschlagen, dass das Steuergerät zur Ausführung des erfindungsgemäßen Verfahrens geeignet ist, wenn das Computerprogramm auf dem Steuergerät, insbesondere 10 auf einer von dem Steuergerät umfassten Recheneinheit, abläuft.

Zeichnungen

15 Weitere Merkmale, Anwendungsmöglichkeiten und Vorteile der Erfindung ergeben sich aus der nachfolgenden Beschreibung von Ausführungsbeispielen der Erfindung, die in den Zeichnungen dargestellt sind. Dabei bilden alle 20 beschriebenen oder dargestellten Merkmale für sich oder in beliebiger Kombination den Gegenstand der Erfindung, unabhängig von ihrer Zusammenfassung in den Patentansprüche oder deren Rückbeziehung sowie unabhängig von ihrer Formulierung beziehungsweise Darstellung in der 25 Beschreibung beziehungsweise in den Zeichnungen. Es zeigen:

30 Figur 1 ein erfindungsgemäßes Steuergerät gemäß einer bevorzugten Ausführungsform zur Steuerung und/oder Regelung einer Sicherheitsanwendung in einem Kraftfahrzeug;

Figur 2 ein Ablaufdiagramm eines erfindungsgemäßen Verfahrens zur Signaturanalyse;

35 Figur 3 ein Steer-by-Wire System mit zwei Steuergeräten;

Figur 4 ein Rechnersystem, das zur Abarbeitung einer in Software realisierten Ausführungsform des erfindungsgemäßen Verfahrens geeignet ist;

5

Figur 5 ein Ablaufdiagramm eines erfindungsgemäßen Signaturbildungsverfahrens gemäß einer bevorzugten Ausführungsform;

10 Figur 6 Funktionsprinzip des MISR-Verfahrens zur Signaturbildung in Hardware; und

15 Figur 7 Funktionsprinzip des MISR-Verfahrens zur Signaturbildung unter Verwendung von EXOR-Operatoren.

Beschreibung der Ausführungsbeispiele

20 Aus dem Stand der Technik ist die Signaturanalyse als eine Methode zur Datenkomprimierung bekannt, die auf der sogenannten CRC (Cyclic Redundancy Check)-Methode basiert. In Abhängigkeit der zu übertragenden Daten wird nach einem vorgebbaren Signaturbildungsverfahren ein resultierendes Datenwort gebildet, welches die Signatur darstellt und den komprimierten Datenstrom charakterisiert. Die Wahrscheinlichkeit, dass trotz Fehlern in dem Datenstrom die richtige Signatur gebildet wird, ist sehr gering und nimmt mit der Datenbreite des Signaturwortes ab. Signaturen 30 werden beispielsweise eingesetzt für die Sicherung von Datenübertragungen, zur Überwachung von Festwertspeichern von Computern und zur Identifizierung von Datenströmen, wie zum Beispiel von digitalen Hardware-Signalen im Rahmen eines Tests von Schaltungen oder von Daten, die ein 35 Schriftstück enthält.

In Hardware wird die Signatur üblicherweise sequentiell Bit für Bit mit Hilfe eines linear rückgekoppelten Schieberegisters (LFSR; Linear Feedback Shift Register) gebildet. Es ist bekannt, dieses sequentielle Signaturanalyse-Verfahren auch in Software zu realisieren. Dieses Verfahren ist aber wegen der sequentiellen Arbeitsweise entweder sehr langsam oder braucht relativ viel Speicherplatz für Tabellen (sogenannte Lookup-Tables), um die Berechnungszeit zu verkürzen.

Des weiteren ist ein Verfahren zur Signaturbildung mit einer parallelen Arbeitsweise (Datenwort für Datenwort) bekannt. Dieses bekannte Verfahren arbeitet beispielsweise mit Hilfe eines Signaturregisters mit mehreren Eingängen (MISR; Multiple Input Shift Register). Durch die parallele Arbeitsweise können zum Beispiel mit einem 32 Bit-Prozessor 32 Bit Eingangsdaten in einem Schritt verarbeitet werden.

Gerade für Sicherheitsanwendungen in einem Kraftfahrzeug, wie beispielsweise Steer-by-Wire oder autonomes Fahren, werden fehlertolerante und damit redundante Elektroniksysteme eingesetzt; die einen hohen Bedarf an sicherem Datenaustausch zwischen Systemen, Steuergeräten und Recheneinheiten haben. Da bei diesen Systemen neben der Rechenzeit die Datenkommunikationszeit einen Engpass darstellt, bietet eine Verwendung des MISR-Verfahren zur Absicherung der vielen Datenkommunikationen einen entscheidenden Vorteil. Es reduziert die erforderliche Rechenzeit zur Absicherung der Datenkommunikation bei Sender und Empfänger auf ein Minimum. Bei den bekannten Verfahren kann es vorkommen, dass trotz einer fehlerhaften Datenübertragung in der Empfangseinheit eine fehlerfreie Signatur gebildet wird. Dies wird als Fehlermaskierung bezeichnet.

Dem gegenüber hat das erfindungsgemäße Verfahren den Vorteil, dass die Wahrscheinlichkeit einer Fehlermaskierung deutlich reduziert wird.

5

In Figur 1 ist ein Steuergerät SG1 zum Steuern und/oder Regeln einer Sicherheitsanwendung 40 dargestellt. Das erfindungsgemäße Verfahren ist beispielsweise als ein Computerprogramm realisiert, das auf den Recheneinheiten

10 RE1, RE3 des Steuergeräts SG1 abläuft. Das Computerprogramm ist auf Speicherbereichen 21, 31 von Speicherelementen 20, 30 abgespeichert, wobei die Speicherbereiche 21, 31 als Nur-Lese-Speicher (ROM; Read Only Memory) ausgebildet sind. Zur Ausführung des Computerprogramms wird dieses über die 15 Datenverbindungen 25, 35 an die Recheneinheiten RE1, RE3 übertragen. Ergebnisse von Berechnungen, die im Rahmen der Abarbeitung des Computerprogramms ermittelt werden, werden in umgekehrter Richtung über die Datenleitungen 25, 35 an die Speicherelemente 20, 30 übertragen und dort in den 20 Speicherbereichen 22, 32 abgespeichert, die als Schreib-Lese-Speicher (RAM; Random Access Memory) ausgebildet sind.

Eine mögliche Arbeitsweise des Steuergeräts SG1 ist folgende:

Über die Datenleitung 45 erhält das Steuergerät SG1 5 Eingangsgrößen, die den Zustand der zu steuernden und/oder zu regelnden Sicherheitsanwendung 40 beschreiben. Diese Eingangsgrößen werden mindestens einer Recheneinheit RE1

30 zugeführt. Dort wird durch eine Ausführung des in dem Speicherbereich 21 abgespeicherten Computerprogramms eine erste Signatur der die Eingangsgrößen beschreibenden Daten erzeugt. Diese Daten werden dann zusammen mit der erzeugten ersten Signatur über die Datenleitung 15 an die 35 Recheneinheit RE3 übertragen. Dort wird anhand der

erhaltenen Daten eine zweite Signatur erzeugt und diese mit der von der Recheneinheit RE1 erhaltenen ersten Signatur verglichen. Stimmen beide Signaturen überein, so wird davon ausgegangen, dass die von der Recheneinheit RE1

5 Übertragenen Daten und die von der Recheneinheit RE3 erhaltenen Daten übereinstimmen.

In einer anderen möglichen Arbeitsweise des Steuergeräts SG1 werden die Eingangsgrößen direkt beiden Recheneinheiten

10 RE1, RE3 zugeführt. Beide Recheneinheiten RE1, RE3 bilden daraufhin eine jeweils erste Signatur, die die erhaltenen Eingangsgrößen charakterisieren. Über die Datenleitung 15 werden daraufhin die Signaturen zwischen den Recheneinheiten RE1, RE3 ausgetauscht. Die so empfangenen

15 Signaturen werden dann mit der selbst erzeugten ersten Signatur in jeder der beiden Recheneinheiten RE1, RE3 verglichen. Stellt mindestens eine der Recheneinheiten RE1, RE3 fest, dass die selbst erzeugte erste Signatur und die von der jeweils anderen Recheneinheit empfangene Signatur

20 nicht übereinstimmen, so veranlasst diese Recheneinheit, dass die entsprechende Anwendung oder das gesamte Steuergerät SG1 in einen definierten Ruhezustand versetzt wird (fail silent). Ein solcher Fehlerfall kann

5 beispielsweise durch eine Störung der Datenleitung 15 auftreten oder durch eine Störung einer nicht dargestellten Datenleitung mit zugehöriger Logik, die die Eingangsgrößen innerhalb des Steuergerätes SG1 von einem Dateneingang zu den Recheneinheiten RE1, RE3 übermittelt.

30 Stimmen die Signaturen der Eingangsgrößen in beiden Recheneinheiten RE1, RE3 überein, wo werden diese Eingangsgrößen entsprechend der Steuer- und Regelaufgabe des Steuergerätes SG1 weiterverarbeitet. Dabei werden Ergebnisdaten erzeugt, die zur Steuerung und/oder Regelung 35 der Sicherheitsanwendung 40 benötigt werden.

Anhand der in den Recheneinheiten RE1, RE3 erzeugten Ergebnisdaten wird in den Recheneinheiten RE1, RE3 jeweils eine erste Signatur erzeugt. Diese ersten Signaturen werden 5 dann über die Datenleitung 15 mit der jeweils anderen Recheneinheit ausgetauscht. Jede Recheneinheit RE1, RE3 vergleicht daraufhin die so erhaltene Signatur mit der zuvor berechneten ersten Signatur. Stimmen beide Signaturen nicht überein, so wird davon ausgegangen, dass ein 10 Übertragungsfehler der Signaturen oder ein Berechnungsfehler der Ergebnisdaten in mindestens einer Recheneinheit vorliegt und es werden entsprechende Maßnahmen (bspw. Neuberechnung oder fail silent) veranlasst.

15 Stimmen die selbst berechnete und die übertragene Signatur bei beiden Recheneinheiten RE1, RE3 überein, so veranlasst das Steuergerät SG1, dass diese Ergebnisdaten zur Steuerung der Sicherheitsanwendung 40 an diese mittels der 20 Datenleitung 45 übertragen werden. Ebenfalls wird die dazugehörige Signatur als eine erste Signatur an die Sicherheitsanwendung 40 übermittelt. Dabei kann die Signatur als eigenständige Botschaft übermittelt werden oder über mehrere Botschaften verteilt werden, 25 beispielsweise, wenn die durch die Signatur charakterisierten Ergebnisdaten selbst auch über mehrere Botschaften verteilt übermittelt werden. Somit wird das Datenvolumen der zu übertragenden Daten verringert und eine Störanfälligkeit der übertragenden Daten reduziert.

30 Figur 2 zeigt ein Ablaufdiagramm eines erfindungsgemäßen Verfahrens zur Signaturanalyse. Hierbei wird davon ausgegangen, dass eine Recheneinheit RE1 eines Steuergerätes SG1 einer anderen Recheneinheit RE2, die 35 Bestandteil eines anderen Steuergerätes SG2 ist, Daten

übermittelt und in der Recheneinheit RE2 mittels einer Signaturanalyse geprüft werden soll, ob ein Übertragungsfehler vorliegt.

- 5 Dazu wird in einem Schritt 90 das Verfahren gestartet. In einem Schritt 91 erzeugt die erste Recheneinheit RE1 in Abhängigkeit der zu übermittelnden Daten D eine erste Signatur S. Dazu wird ein vorgebares Signaturbildungsverfahren SBV verwendet. In einem Schritt 10 92 werden diese Daten zusammen mit der ersten Signatur S mittels geeigneter Botschaften an die zweite Recheneinheit RE2 übermittelt. Dort wird in einem Schritt 93 anhand der übermittelten Daten D' ebenfalls gemäß des Signaturbildungsverfahrens SBV eine zweite Signatur S' 15 erzeugt. In einem Abfrageschritt 94 werden in der Recheneinheit RE2 die Signaturen S, S' auf Übereinstimmung geprüft. Stimmen diese Signaturen S, S' überein, so wird davon ausgegangen, dass kein Übertragungsfehler vorliegt und das Verfahren in Schritt 96 beendet. Unterscheiden sich 20 jedoch die Signaturen S, S', so wird auf eine Störung der Datenübertragung geschlossen und in einem Schritt 95 ein Verfahren gestartet, das die zugrunde liegende Sicherheitsanwendung in einen definierten Ruhezustand überführt (fail silent). Anschließend endet das Verfahren 25 in Schritt 96.

- Figur 3 zeigt eine Sicherheitsanwendung 40, die mit zwei Steuergeräten SG1, SG2 betrieben wird. Dabei entsprechen die in Figur 1 vorhandenen Elemente den in Figur 3 30 dargestellten Elementen mit denselben Bezugszeichen. In Figur 3 ist zusätzlich das Steuergerät SG2 gezeigt, das zwei Recheneinheiten RE2, RE4 umfasst, die über eine Datenleitung 515 miteinander verbunden sind. Die Recheneinheit RE2 ist mittels einer Datenleitung 525 mit 35 einem Speicherelement 520 und die Recheneinheit RE4 ist

mittels einer Datenleitung 535 mit einem Speicherelement 530 verbunden. Außerdem ist die Recheneinheit RE1 des Steuergerätes SG1 mit der Recheneinheit RE2 des Steuergerätes SG2 über die Datenleitung 46 verbunden.

5 Analog dazu verbindet die Datenleitung 47 die Recheneinheit RE3 mit der Recheneinheit RE4.

Ein Anwendungsbeispiel ist ein Steer-by-Wire System, bei dem mittels eines an einem Lenkrad 60 angebrachten

10 Absolutwinkelgebers 61 ein Lenkwunsch eines Fahrers erkannt und mittels einer Datenleitung 45 dem Steuergerät SG1 zugeführt wird und durch das zweite Steuergerät SG2 eine hydraulische Lenkvorrichtung 70 gesteuert wird, die ein Einlenken der Räder 71 bewirkt.

15 Für die sichere Datenübertragung zwischen den Steuergeräten SG1, SG2 sind die beiden redundanten Datenleitungen 46, 47 beispielsweise als CAN, TTCAN oder FlexRay realisiert. Für eine Kommunikation zwischen Steuergerät SG1 und Steuergerät 20 SG2 werden pro Regelzyklus mehrere Botschaften ausgetauscht, die beispielsweise Messgrößen, Stellgrößen und Statusinformationen enthalten.

5 Wird über die Datenleitung 45 ein Lenkbefehl übermittelt, so wird mit den dadurch übermittelten Daten wie in dem in Figur 1 beschriebenen Verfahren in den Recheneinheiten RE1, RE3 des Steuergerätes SG1 eine Signaturanalyse durchgeführt. Anschließend werden die Daten entsprechend von in den Speicherelementen 20, 30 abgespeicherten 30 Computerprogrammen verarbeitet. Die damit erzeugten Ergebnisdaten werden wieder der bereits in Figur 1 dargestellten Signaturanalyse bezüglich der Recheneinheiten RE1, RE3 unterzogen. Wird dabei kein Übertragungs- oder Rechenfehler erkannt, so übermitteln die Recheneinheiten 35 RE1, RE3 mittels der redundant ausgelegten Datenleitungen

46, 47 die Ergebnisdaten an die Recheneinheiten RE2, RE4 des Steuergeräts SG2. Dabei wird auch hier zusammen mit den Daten die entsprechende erste Signatur übermittelt.

- 5 Die Recheneinheiten RE2, RE4 bilden nun ihrerseits unabhängig voneinander anhand der jeweils erhaltenen Daten eine zweite Signatur und vergleichen diese mit der übermittelten ersten Signatur, um eine Störung der Daten während der Übertragung von dem Steuergerät SG1 zu dem
- 10 Steuergerät SG2 zu erkennen.

In Figur 4 ist ein Rechnersystem 2 schematisch dargestellt, das eine Recheneinheit RE1 und einen Nur-Lese-Speicher 21 umfasst. Die Recheneinheit RE1 umfasst eine sogenannte ALU (Arithmetic-Logic-Unit) 80 und mehrere Register FR, Rom_Adress, Sx, Dx, Gx zur Speicherung von Adressen und Daten. Außerdem verfügt die Recheneinheit RE1 über ein sogenanntes Carry-Flag CF, das durch einen 1-bit breiten Bereich des Flag-Registers FR realisiert ist. Über eine Adressleitung 25a und eine Datenleitung 25b sind die Recheneinheit RE1 und der Nur-Lese-Speicher 21 verbunden. Der Nur-Lese-Speicher 21 umfasst adressierbare Bereiche zur Speicherung von Daten, von denen aus Gründen der Übersichtlichkeit in Figur 4 nur drei Bereiche dargestellt und mit den Bezugszeichen A1, A2, A3 versehen sind.

Eine auf dem Rechnersystem 2 ablaufende Signaturbildung, eines erfindungsgemäßen Verfahrens, wie es beispielsweise in den Figuren 1, 3 dargestellt ist, ist mittels eines Ablaufdiagramms in Figur 5 beschrieben. Das Verfahren beginnt in einem Schritt 100. In einem Schritt 101 wird ein Adressregister Rom_Adress mit einer Adresse geladen, die auf einen adressierbaren Bereich A1 des Nur-Lese-Speichers 21 verweist. In einem Schritt 102 wird ein Signaturregister Sx mit dem Wert null initialisiert. Das Verfahren ist so

ausgelegt, das sich bei Beenden des Verfahrens in dem Signaturregister Sx die berechnete Signatur befindet. In einem Schritt 103 wird ein Generatorpolynom, das geeignet ist, über einer Menge von Daten eine diese Daten 5 charakterisierende Signatur zu bilden, in das Register Gx geladen.

In einem Schritt 104 wird der Speicherinhalt A1, der sich unter der in dem Adressregister Rom_Adr gespeicherten 10 Adresse in dem Nur-Lese-Speicher 21 befindet, in das Register Dx der Recheneinheit RE1 geladen. Dazu wird die in dem Register Rom_Adr gespeicherte Adresse, die beispielsweise auf den Speicherbereich A1 verweist, über die Adressleitung 25a dem Nur-Lese-Speicher 21 übermittelt. 15 Dieser sendet dann die sich in dem entsprechenden Speicherbereich A1 befindlichen Daten mittels der Datenleitung 25b an die Recheneinheit RE1, wo sie in dem Register Dx abgelegt werden. Das Adressregister Rom_Adr wird daraufhin in einem Schritt 105 um eine Datenadresse 20 erhöht.

In einem Schritt 106 wird der Inhalt des Signaturregisters Sx um eine Stelle (entspricht einem Bit) nach links verschoben (left-shift). Dabei wird eine Schreibweise für 5 Binärdaten zugrunde gelegt, bei der das höchstwertige Bit, das sogenannte MSB (Most Significant Bit), stets links notiert wird. Dieser left-shift innerhalb des Signaturregisters Sx bewirkt, dass das MSB aus dem entsprechenden Datenwort herausfällt und in dem Carry-Flag 30 CF abgelegt wird.

In einem Schritt 107 wird mittels der ALU 80 ein EXOR-Operator auf das Datenregister Dx und das Signaturregister Sx angewendet und das Ergebnis in dem Signaturregister Sx 35 abgelegt. In einem Abfrageschritt 108 wird überprüft, ob

der in dem Carry-Flag CF abgelegte Wert MSB gleich 1 ist. Ist dies nicht der Fall, so wird zu einem Schritt 110 verzweigt. Gilt aber MSB = 1, so wird in einem Schritt 109 das in dem Register Gx abgelegte Generatorpolynom von dem 5 aktuellen Wert des Signaturregisters Sx subtrahiert, was mittels einer EXOR-Operation in der ALU 80 erreicht wird.

In dem folgenden Schritt 110 wird nun der Inhalt des Datenregisters Dx invertiert. Daraufhin wird in dem Schritt 10 111 der Inhalt des Signaturregisters Sx erneut um ein Bit nach links geschoben und das herausfallende Bit wird dabei als MSB in dem Carry-Flag CF abgelegt, wobei der bisher dort gespeicherte Wert überschrieben wird. In Schritt 112 wird mittels der ALU 80 der EXOR-Operator wieder auf das 15 Datenregister Dx und das Signaturregister Sx angewendet und das Ergebnis in dem Signaturregister Sx abgelegt.

In einem Abfrageschritt 113 wird analog zu Schritt 108 überprüft, ob der in dem Carry-Flag CF abgelegte Wert MSB 20 gleich eins ist. Ist dies nicht der Fall, so wird zu einem Abfrageschritt 115 verzweigt. Gilt aber MSB = 1, so wird in einem Schritt 114 das in dem Register Gx abgelegte Generatorpolynom von dem aktuellen Wert des Signaturregisters Sx subtrahiert, was wieder mittels der EXOR-Operation in der ALU 80 erreicht wird.

In dem Abfrageschritt 115 wird überprüft, ob ein Ende des Datenwortes, über das die Signatur gebildet wird, bereits erreicht ist. Ist dies nicht der Fall, so wird das 30 Verfahren an dem Schritt 104 fortgesetzt. Ist das Ende des Datenwortes erreicht, so endet das Verfahren, wobei sich nun in dem Signaturregister Sx die erzeugte Signatur befindet.

35 Das erfindungsgemäß vorgeschlagene Software-Verfahren zur

Signaturbildung arbeitet nach dem sogenannten MISR (Multiple-Input Signature Register)-Verfahren. Es bildet das an sich in Hardware bekannte Verfahren in Software ab.

5 In Figur 6 ist ein Funktionsprinzip des MISR-Verfahrens in Hardware für Datenworte von 5 Bit Breite (D_0 bis D_4) dargestellt. SRB0 bis SRB4 stellen 5 Bitspeicher (FlipFlops) des rückgekoppelten Schieberegisters zur Signaturbildung dar. An jedem Bitspeicher-Eingang sitzt ein
 10 Modulo-2-Addierer (+) mit zwei oder drei Eingängen. An dem ersten Eingang des Addierers liegt ein Bit des zu komprimierenden Datenwortes D (D_i), an dem zweiten Eingang liegt der Ausgang des davor liegenden Bitspeichers SRB (SRB_{i-1}) und an dem dritten Eingang liegt gegebenenfalls der
 15 Ausgang des höchsten Bitspeichers SRB4 des Schieberegisters.

In dem vorliegenden Fall, indem die Datenworte binär codiert sind, wird durch einen Modulo-2-Addierer der Rest der Division $(D_i + SRB_{i-1} + SRB_4) / 2$ beziehungsweise der Division $(D_i + SRB_{i-1}) / 2$ gebildet. Das gesamte Signaturregister S_x bildet ebenfalls einen Restwert. Der gesamte Eingangsdatenblock, der aus Datenworten mit einer Länge von 5 Bit besteht, kann vereinfacht als ein sequentieller Datenstrom betrachtet werden, der durch einen Wert, das Divisor-Polynom, dividiert wird. Der Rest dieser Division befindet sich anschließend in dem Signaturregister S_x .

30 Mathematisch kann die Arbeitsweise von Signaturregistern durch die Polynomdarstellung von Binärzahlen beschrieben werden: Für die Polynomdarstellung einer Binärzahl werden die 1-Ziffern in der Binärzahl durch entsprechende Terme x^n ersetzt, wobei der Exponent n der Stellen-Wertigkeit der
 35 entsprechenden 1 entspricht. Dem Divisorpolynom

$G(x) = x^5 + x^3 + x^1 + x^0$ in Figur 6 entspricht die Binärzahl 101011.

Da ein Divisorpolynom außer zur Datenkomprimierung auch zur Datengeneration verwendet werden kann, wird das Divisorpolynom auch Generatorpolynom $G(x)$ genannt. Ebenso

5 kann der Eingangsdatenstrom als Polynom $D(x)$ und der Inhalt des Signaturregisters als Polynom $S(x)$ dargestellt werden.

In dem Signaturregister Sx wird die Division durch das Divisorpolynom durch wiederholte Subtraktion des

10 Divisorpolynom-Wertes von dem Schieberegister-Inhalt vorgenommen. Jedes mal wenn aus dem höchsten Bitspeicher SRB4 der Wert 1 herausgeschoben wird und somit in einem längeren Schieberegister der Term x^5 in dem nicht vorhandenen Bitspeicher SRB5 (entsprechend dem Carry-Flag

15 CF in Figur 4, 5) gesetzt würde, wird der in dem Schieberegister gespeicherte Wert um $x^5 + x^3 + x^1 + x^0$ verringert.

Das Herausschieben aus SRB4 verringert um x^5 , die Rückkopplung über die Modulo-2-Addierer verringert den Schieberegister-Wert um $x^3 + x^1 + x^0$, weil die Modulo-2-Addition 20 in diesem Fall einer Modulo-2-Subtraktion entspricht. Die Hardware-Darstellung des Divisorpolynoms wird durch die Rückkopplungspfade von dem höchsten Bit des Schieberegisters zu den Modulo-2-Addierern gebildet.

In Figur 6 sind die Modulo-2-Addierer vor folgenden Bitspeichern mit SRB4 verbunden: SRB3, SRB1, SRB0. Das zugehörige Polynom lautet $x^5 + x^3 + x^1 + x^0$. Der Term x^5 mit dem Exponent 5 (entsprechend der Länge des Schieberegisters + 1) ist enthalten, weil das höchstwertige Bit MSB

30 hinausgeschoben und damit subtrahiert wird. Mathematisch lässt sich zeigen, dass der in dem Signaturregister verbleibende Rest dem Rest entspricht, der durch die Division des Datenstroms $D(x)$ durch das Generatorpolynom $G(x)$ entsteht.

Nachfolgend wird mittels einer rekursiven Funktion gezeigt, welche mathematischen Operationen zur Bildung der Signatur $S(x)$ in dem Signaturregister erforderlich sind. Es bezeichnet:

5

$S_i(x)$: aktueller Wert des Signaturregisters;
 $S_{i+1}(x)$: nächster Wert des Signaturregisters nach der Verarbeitung des Datenworts; und
 $D_i(x)$: aktueller Wert des Datenregisters.

10

Die Signatur nach dem i -ten Verarbeitungszyklus ist dann:

$$S_{i+1}(x) = [D_i(x) + xS_i(x)] \bmod G(x)$$

15 Die Multiplikation der Signatur $S_i(x)$ mit x stellt das Schieben des Schieberegisters um eine Bitposition nach links dar (left shift). Das Zeichen „+“ ist eine Modulo-2-Addition beziehungsweise Subtraktion des Signaturregisters mit dem aktuellen Datenwort. Der Begriff $\bmod G(x)$ bedeutet,
20 dass in dem Signaturregister immer der Rest bezüglich der Division $S(x)/G(x)$ steht. Dies wird in der Hardware-Lösung, wie schon beschrieben, dadurch erreicht, dass immer dann, wenn das höchste Bit in dem Schieberegister = 1 ist, das Generatorpolynom $G(x)$ von dem Wert $S_i(x)$ des Schieberegisters abgezogen wird.

25

Bei einer Software-Realisierung des MISR-Verfahrens, wie es beispielsweise in Figur 5 dargestellt ist, können die Modulo-2-Addierer gemäß Figur 7 als EXOR-Operationen 30 realisiert sein. An den Stellen, wo zwei Bits zu addieren sind, ist eine EXOR-Operation vorgesehen, an den Stellen, wo drei Bits zu addieren sind, wird dies durch zwei aufeinanderfolgende EXOR-Operationen durchgeführt. Die Hardware-Darstellung kann in Software umgesetzt werden, 35 indem folgende Schritte für jedes zu verarbeitende

Datenwort D durchgeführt werden.

1. Der Anfangsinhalt des Signaturregisters sei 00000.
2. Schiebe den Inhalt des Signaturregisters um eine Position nach links. Mathematisch bedeutet dies die Multiplikation des Signaturregisterwertes mit x, also $S_{i+1}(x) = xS_i(x)$.
- 5 3. Bilde EXOR aus aktuellem einzulesendem Datenwort $D_i(x)$ und dem Inhalt des Signaturregisters $S(x)$, also $S_{i+1}(x) = D_i(x) + xS_i(x)$.
- 10 4. Falls SRB4=1: Bilde EXOR aus Signaturregister $S(x)$ und Generatorpolynom $G(x)$. Damit ist $S_{i+1}(x) = S_{i+1}(x) + G(x) = [D_i(x) + xS_i(x)] \bmod G(x)$.

15 Die beschriebenen Signaturbildung über einen ROM-Speicher kann auf jedem beliebigen Rechengerät realisiert werden. Wie bereits erwähnt, werden zur Realisierung dieses Signaturbildungsverfahrens auf einem Prozessor vier Register benötigt:

20

- ROM_Adr als Adressregister für das ROM;
- Dx als Register für das aktuelle zu verarbeitende Datenwort;
- Sx als Signaturregister; und
- Gx als Register für die Speicherung des Generatorpolynoms.

Der symbolische Programmablauf in Assembler ist dann wie folgt:

30

- Start: Lade ROM_Adr mit Anfangsadresse des ROM
- Init: Lade Sx mit 0 ($S_0(x) = 0$)
- Lade Gx mit Generatorpolynom
- Loop: Lade Dx mit Inhalt von ROM_Adr
- 35 Erhöhe ROM_Adr um 1

MISR: Schiebe S_x um 1 Bit nach links, höchstes Bit in
Carry-Bit ($xS(x)$)
Bilde EXOR D_x mit S_x ($D(x) + xS(x)$)
Wenn Carry-Bit gesetzt, bilde EXOR S_x mit G_x
([$D(x) + xS(x)$]mod $G(x)$)
Vergleiche ROM_Adr mit Endadresse ROME
Wenn ungleich, gehe zu Loop

5 07.11.2002

1. Robert Bosch GmbH, 70442 Stuttgart
2. ZF Lenksysteme GmbH, Richard-Bullinger-Str. 77
73527 Schwäbisch Gmünd

10

Ansprüche

1. Verfahren zur Übertragung von Daten (D), bei dem nach einem vorgebbaren Signaturbildungsverfahren (SBV) eine erste Signatur (S) in Abhängigkeit der zu übertragenden Daten (D) gebildet, die erste Signatur (S) zusammen mit den Daten (D) in sogenannten Botschaften übertragen, nach dem Signaturbildungsverfahren eine zweite Signatur (S') in Abhängigkeit von den übertragenen Daten (D') gebildet und

20 die erste Signatur (S) mit der zweiten Signatur (S') verglichen wird, dadurch gekennzeichnet, dass die zu übertragenden Daten (D) invertiert werden, nach dem vorgebbaren Signaturbildungsverfahren (SBV) in Abhängigkeit der zu übertragenden Daten (D) und der invertierten Daten (D_i) die erste Signatur (S) gebildet wird, die erste Signatur (S) und die Daten (D) übertragen werden, die übertragenen Daten (D') invertiert werden, in Abhängigkeit dieser invertierten Daten (D'_i) und der übertragenen Daten (D') nach dem Signaturbildungsverfahren (SBV) die zweite Signatur (S') gebildet wird und die erste Signatur (S) mit der zweiten Signatur (S') verglichen wird.

25 30
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass mittels eines Signaturregisters (S_x) mit mehreren Eingängen (MISR; Multiple Input Shift Register) die erste

Signatur (S) und/oder die zweite Signatur (S') bit-parallel (wortweise) gebildet werden.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass die erste Signatur (S) und/oder die zweite Signatur (S') über mehrere Botschaften gebildet werden.
4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, dass eine Signatur (S) auf mehrere Botschaften verteilt übertragen wird.
5. Verfahren nach einem der Ansprüche 1 bis 4, wobei die zu übertragenden Daten (D) bitgenaue Eingangsdaten sind, die beispielsweise in Botschaften über Datenbusse zu den Recheneinheiten gelangen, oder Berechnungsergebnisse sind, die parallel auf mehreren Rechnern redundant erzeugt werden, dadurch gekennzeichnet, dass zur Überprüfung einer Übereinstimmung dieser Daten (D) lediglich die entsprechenden Signaturen (S) übertragen werden.
6. Verwendung eines Verfahrens nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, dass das erfindungsgemäße Verfahren zur Überprüfung des Inhaltes eines Speicherbereiches eines Nur-Lese-Speichers (21), Flash-Speichers oder eines Schreib-Lese-Speichers (22) eingesetzt wird.
7. Verwendung nach Anspruch 6 dadurch gekennzeichnet, dass die Daten (D) des zu überprüfenden Speicherinhaltes invertiert werden, nach dem vorgebbaren Signaturbildungsverfahren (SBV) in Abhängigkeit der zu überprüfenden Daten (D) und der invertierten Daten (D_i) eine erste Signatur (S) gebildet und in einem Speicherbereich eines Nur-Lese-Speichers (21), Flash-Speichers oder eines Schreib-Lese-Speichers (22) als eine Sollsignatur abgespeichert wird und dass für eine

Überprüfung der sich in dem zu überprüfenden Speicherbereich befindlichen Daten (D'), die Daten (D') invertiert werden, in Abhängigkeit dieser invertierten Daten (D_i') und der Daten (D') nach dem

5 Signaturbildungsverfahren (SBV) die zweite Signatur (S') gebildet wird und mit der Sollsignatur (S) verglichen wird.

8. Computerprogramm, das auf einem Rechengerät oder einem Steuergerät (SG1, SG2), insbesondere auf einer Recheneinheit (RE1, RE2, RE3, RE4), ablauffähig ist, **dadurch gekennzeichnet**, dass das Computerprogramm zur Ausführung eines Verfahrens nach einem der Ansprüche 1 bis 5 geeignet ist, wenn es auf einer Recheneinheit (RE1, RE2, RE3, RE4) abläuft.

15 9. Computerprogramm nach Anspruch 8, dadurch gekennzeichnet, dass das Computerprogramm auf einem Speicherelement ((20, 30), insbesondere auf einem Schreib-Lese-Speicher mit wahlfreiem Zugriff (RAM; Random-Access-Memory) (22, 32), einem Nur-Lese-Speicher (ROM; Read-Only-Memory) (21, 31) oder einem Flash-Speicher (Flash-Memory), abgelegt ist.

20 10. Steuergerät (SG1) für ein Kraftfahrzeug, umfassend mindestens eine Recheneinheit (RE1) und ein Speicherelement (20), auf dem ein Computerprogramm abgelegt ist, das auf der Recheneinheit RE1 ablauffähig ist, **dadurch gekennzeichnet**, dass das Steuergerät (SG1) zur Ausführung eines Verfahrens nach einem der Ansprüche 1 bis 5 geeignet ist, wenn das Computerprogramm auf der Recheneinheit (RE1) abläuft.

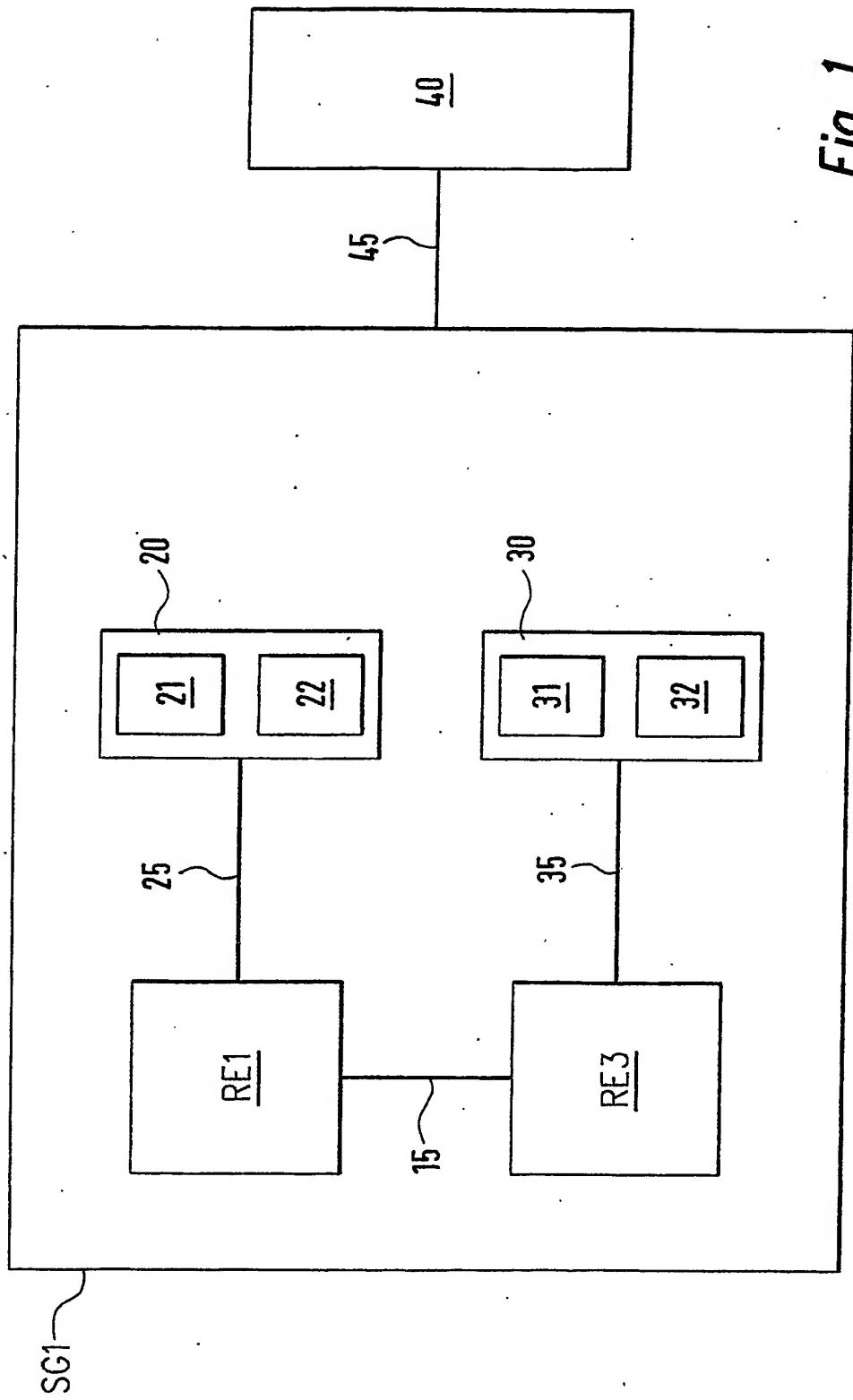


Fig. 1

2 / 7

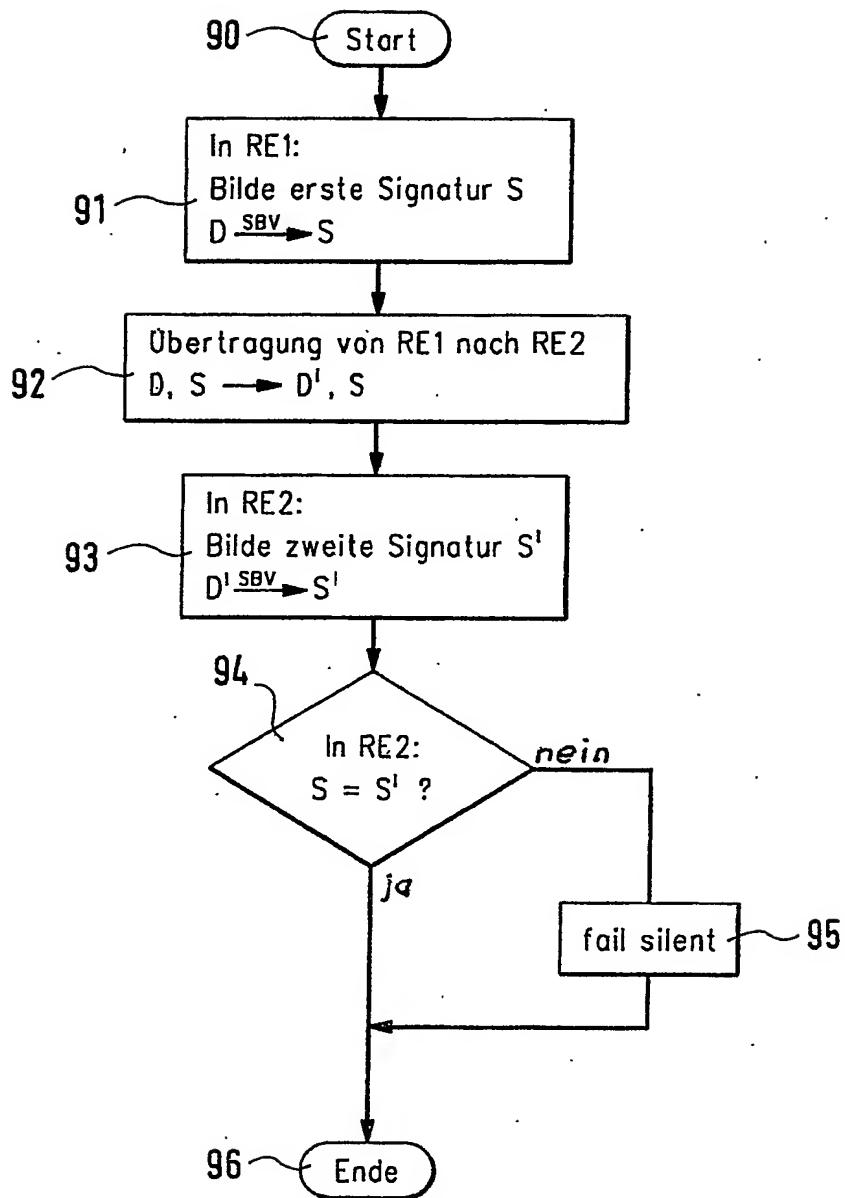


Fig. 2

3 / 7

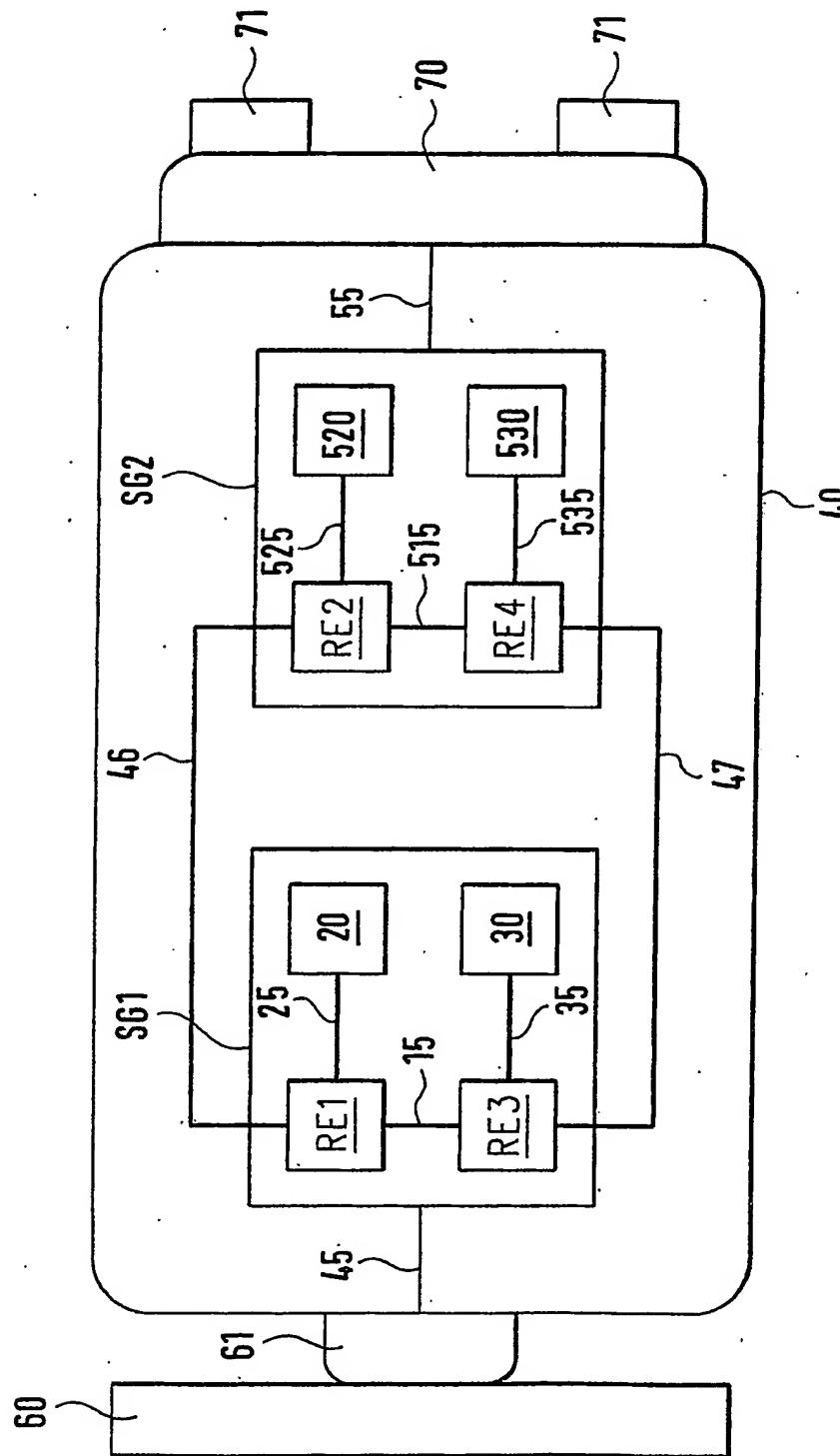


Fig. 3

4 / 7

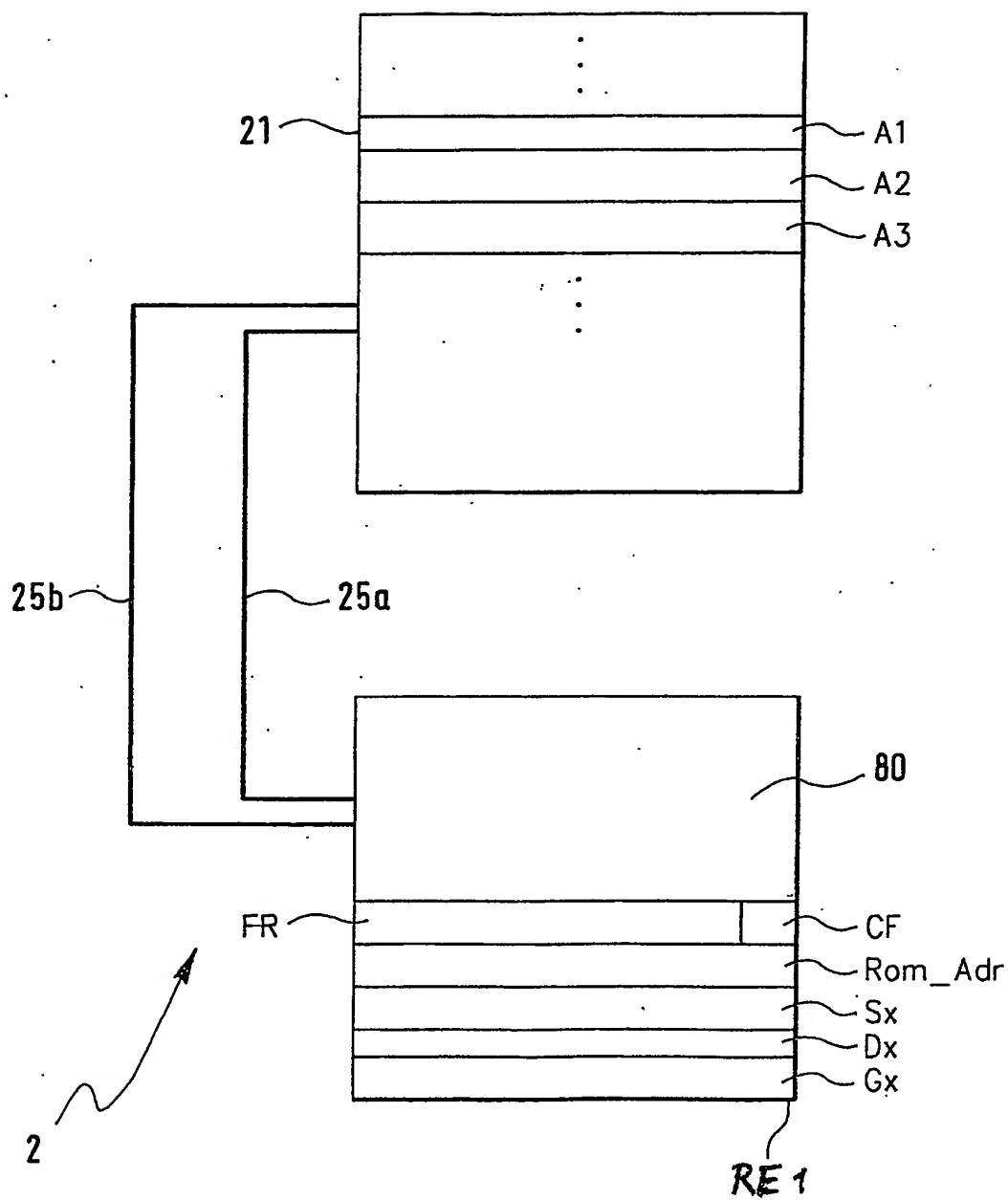


Fig. 4

5 / 7

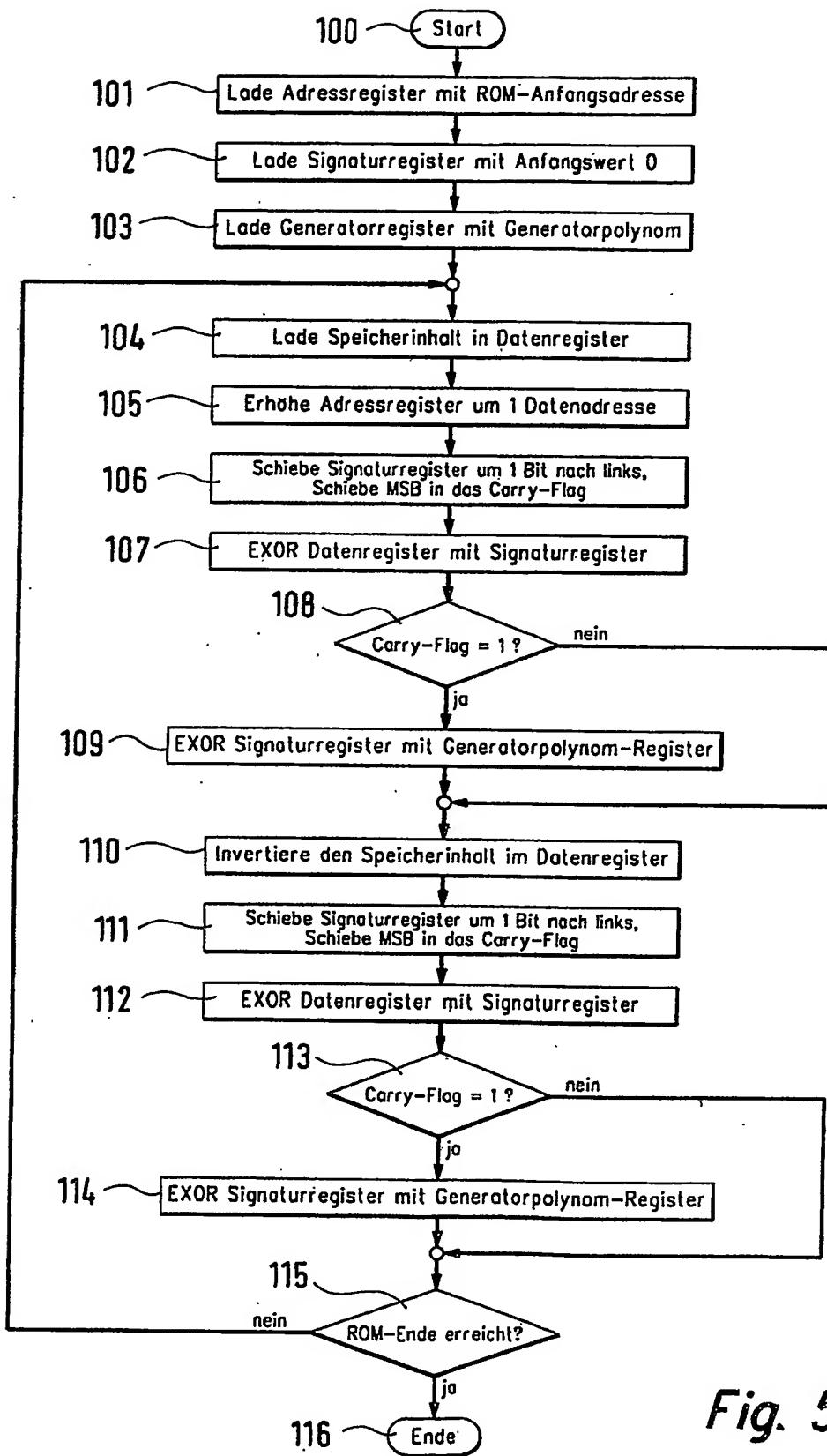


Fig. 5

6 / 7

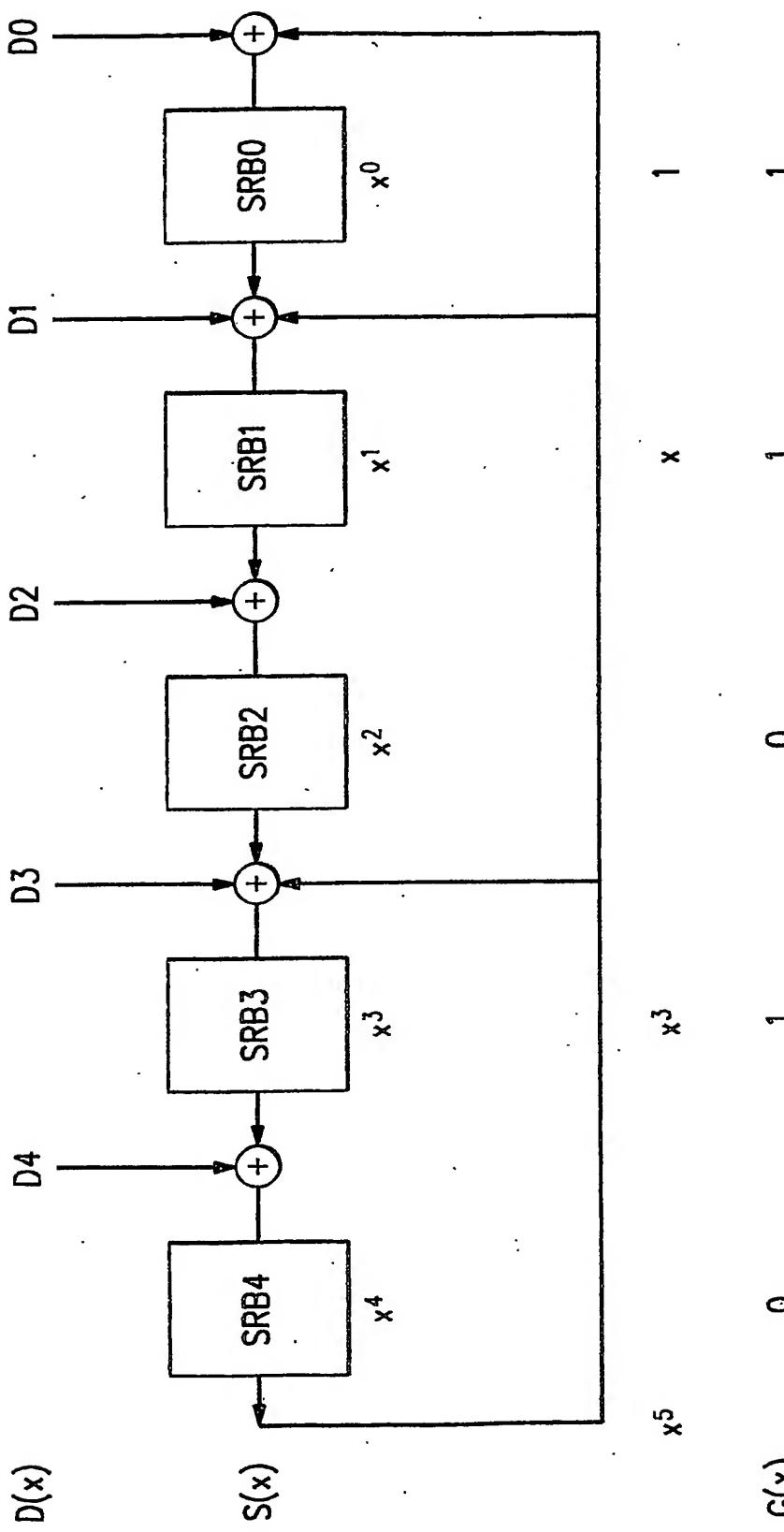


Fig. 6

7 / 7

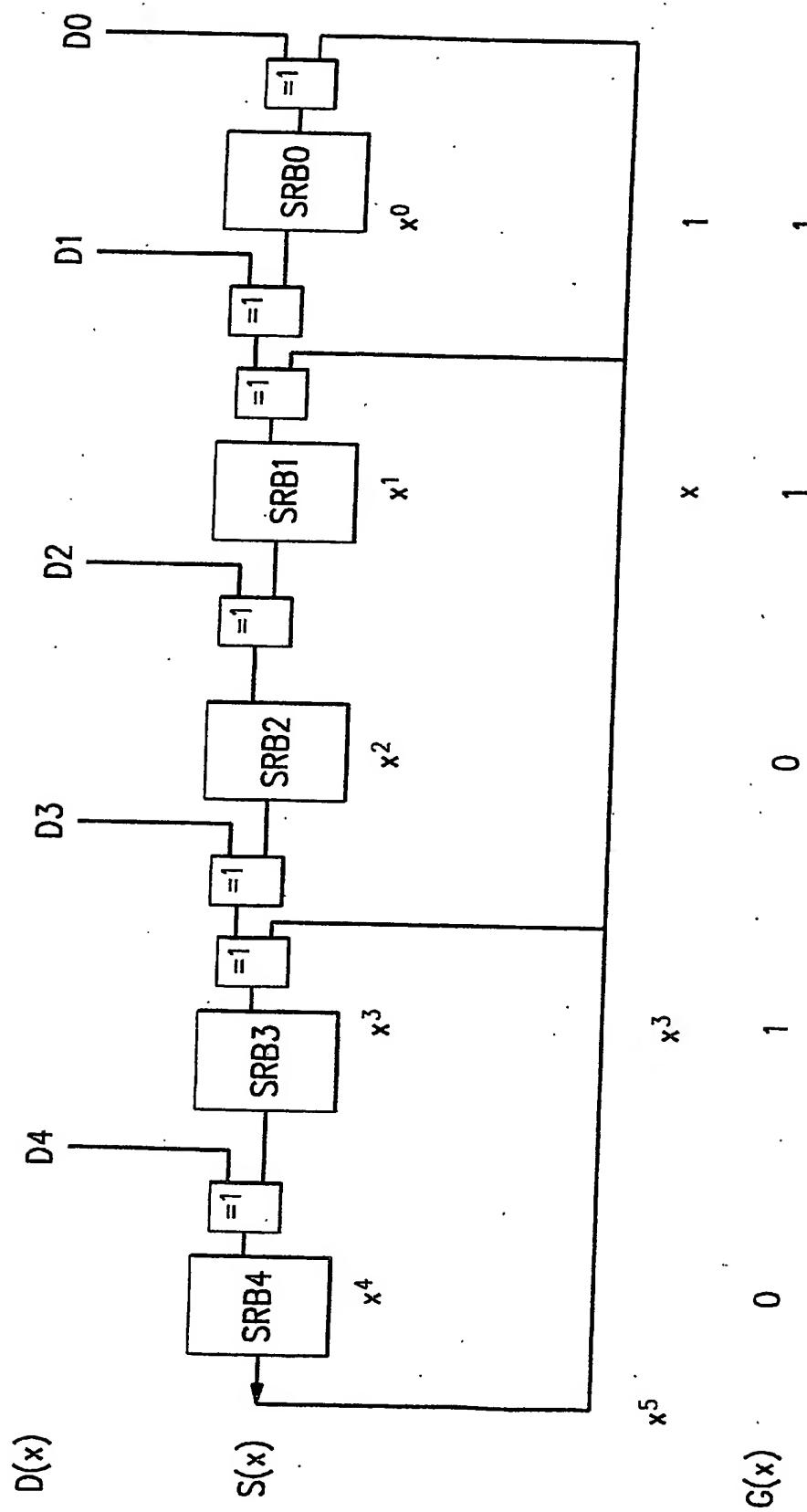


Fig. 7

5 07.11.2002

1. Robert Bosch GmbH, 70442 Stuttgart
2. ZF Lenksysteme GmbH, Richard-Bullinger-Str. 77
73527 Schwäbisch Gmünd

10 Verfahren zur Übertragung von Daten

Zusammenfassung

Die Erfindung betrifft ein Verfahren zur Übertragung von
15 Daten, bei dem nach einem vorgebbaren
Signaturbildungsverfahren (SBV) eine erste Signatur (S) in
Abhängigkeit der zu übertragenden Daten (D) gebildet, die
erste Signatur (S) zusammen mit den Daten (D) übertragen,
nach dem Signaturbildungsverfahren eine zweite Signatur
20 (S') in Abhängigkeit von den übertragenen Daten (D')
gebildet und die erste Signatur (S) mit der zweiten
Signatur (S') verglichen wird. Um bei der Überwachung der
Übertragung der Daten (D) mittels Signaturanalyse die
Fehlermaskierungswahrscheinlichkeit zu reduzieren, wird
vorgeschlagen, dass die zu übertragenden Daten (D)
invertiert werden, nach dem vorgebbaren
Signaturbildungsverfahren (SBV) in Abhängigkeit der zu
übertragenden Daten (D) und der invertierten Daten (D_i)
die erste Signatur (S) gebildet wird, die erste Signatur
30 (S) und die Daten (D) übertragen werden, die übertragenen
Daten (D') invertiert werden, in Abhängigkeit dieser
invertierten Daten (D'_i) und der übertragenen Daten (D')
nach dem Signaturbildungsverfahren (SBV) die zweite
Signatur (S') gebildet wird und die erste Signatur (S) mit
35 der zweiten Signatur (S') verglichen wird. Figur 2

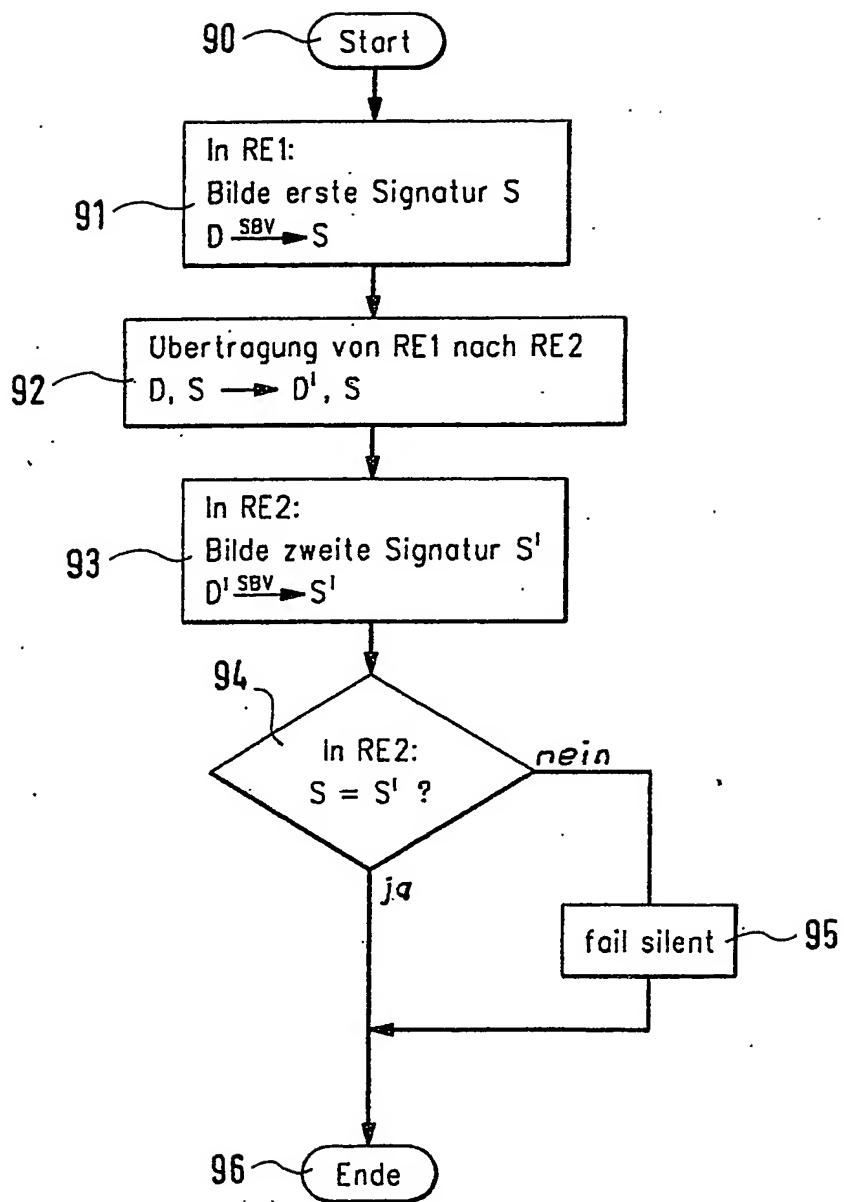


Fig. 2